PRG

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/628,839	07/31/2000	Richard M. Shupak	777.388US1	5501

7590 04/08/2003

John E Whitaker
Merchant & Gould PC
PO Box 2903
Minneapolis, MN 55402-0903

EXAMINER
GROSS, KENNETH A

ART UNIT	PAPER NUMBER
2122	4

DATE MAILED: 04/08/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

PRG

PP4

Office Action Summary	Application No.	Applicant(s)	
	09/628,839	SHUPAK, RICHARD M.	
	Examiner	Art Unit	
	Kenneth A Gross	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE ____ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____ .

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-48* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

Claim Objections

1. Claim 2 is objected to because of the following informalities: the term "an annotation information" should be "annotation information". Appropriate correction is required.

Claim Rejections - 35 USC § 112

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claims 1-13, 45, and 48 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. In regard to Claim 1, Claim 1 recites the limitation "the annotation information" There is insufficient antecedent basis for this limitation in the claim. Claim one further recites "a non-executable statement". The "Microsoft Computer Dictionary: Third Edition" defines "statement" as "*the smallest executable entity within a programming language.*" Therefore, "a non-executable statement" is unclear, and is interpreted to mean "information" or "data". In regard to Claim 10, the term "a second function" is unclear. Specifically, the claim or parent claims make no reference to a *first* function. Claim 13 recites the limitation "the arguments". There is insufficient antecedent basis for this limitation in the claim. In regard to Claims 45 and 48, the term "the computer program analysis tool further comprises a computer program analysis tool" is unclear. Does this term mean that a computer program analysis tool comprises within itself *another* computer program analysis tool?

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claim 19, 20, 22-26, 33, 35, 37, and 43-48 is rejected under 35 U.S.C. 102(b) as being anticipated by Buzbee (U.S. Patent Number 5,815,720).

In regard to Claim 19, Buzbee teaches the following: (a) reading annotation information in an executable computer program (Column 2, lines 6-12); (b) controlling the execution if the first computer analysis tool using the information (Column 2, lines 12-15). Claims 33 and 43 correspond directly with Claim 19 and are rejected for the same reasons as Claim 19.

In regard to Claim 20, Buzbee teaches using the profile information in a compiler that optimizes the code, hence acting as an optimizer.

In regard to Claim 22, Buzbee teaches passing the optimized object code into a translator that places executable profiling code into the source code (Column 9, lines 16-19).

In regard to Claim 23, Buzbee teaches passing the optimized object code into a translator that places executable profiling code into the source code, thus acting as a profiler (Column 9, lines 16-19).

In regard to Claim 24, Buzbee teaches: (a) reading the annotation information in an executable computer program (Column 2, lines 6-12); (b) modifying the executable program in accordance with the information in the annotation program (Column 2, lines 12-15). Claims 35 and 46 correspond with Claim 24 and are rejected for the same reasons as Claim 24.

In regard to Claim 25, Buzbee teaches inserting profiling code into the executable program based on the annotation information (Figure 5, items 42 and 44).

In regard to Claim 26, Buzbee teaches optimizing the executable program using the profile information (Column 9, lines 5-8).

In regard to Claims 37 and 47, Claims 37 and 47 correspond with Claim 25 and are rejected for the same reasons as Claim 25.

In regard to Claim 44, Buzbee teaches the compiler/optimizer first compiles the code without the profile information (Column 8, lines 62-63 and Column 9, lines 5-8). Only with the profile information does the compiler/optimizer optimize the code, otherwise, it would just compile the code normally.

In regard to Claims 45 and 48, Buzbee teaches a computer program analysis tool (Figure 5, item 42), which yields an optimized application that comprises a computer program analysis tool (Figure 5, item 44), which produces profile information.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claim 1-11, 13-15, 17, 18, 28-32, 38, 41, and 42 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kaneshiro et al. (U.S. Patent Number 5,950,003) in view of "Compilers: Principles, Techniques, and Tools" by Alfred Aho (hereinafter Aho).

In regard to Claim 1, Kaneshiro teaches: (a) parsing an annotation representation into the source code (Column 3, lines 51-55); (b) transforming the annotation representation into intermediate language code. Since the program code contains annotations within the code, when the code is compiled (Figure 11, item S21), it will transformed into intermediate language code. Kaneshiro does not teach generating a non-executable statement from the intermediate language code, the annotation information corresponding to the annotation. Aho, however, describes a symbol table that the compiler creates and uses to keep track of variables and functions in the source code. The symbol table keeps track of a variety of arguments of a procedure (Page 11, lines 3-7). For each procedure, a symbol is generated that corresponds to the function. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to pass an annotation representation into the source code and transform the representation into intermediate language code as taught by Kaneshiro, where a symbol is generated for the annotation representation, as taught by Aho, since this allows the compiler to keep track of names generated in the computer program.

In regard to Claim 2, Kaneshiro teaches the method of Claim 1, but does not teach that the annotation information generated contains an address or a plurality of arguments of the annotation representation. Aho, however, describes a symbol table that the compiler creates and

uses to keep track of variables and functions in the source code. The symbol table keeps track of a variety of arguments of a procedure (Page 11, lines 3-7), as well as the procedure name. Aho teaches that a common representation for a name is a pointer, which contains an address of the symbol's location (Page 430, lines 40-41). Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize a method of annotating a computer program with annotation functions as taught by Kaneshiro, where the address and the arguments of the annotation function is kept in a symbol table after compilation, as taught by Aho, since this allows the compiler to keep track of names generated in the computer program.

In regard to Claim 3, Aho teaches that the annotation information regarding the annotation function is kept as a symbol on the symbol table (Page 11).

In regard to Claim 4, Kaneshiro teaches annotation functions with arguments that identify different areas of interest, such as the procedure name being monitored, or the incrementing of a loop (Table 1).

In regard to Claim 5, the annotation functions of Kaneshiro can be interpreted as software components, and when Aho teaches generating arguments for a symbol table, these arguments will be generated according to the software components, or functions, that they are associated with.

In regard to Claim 6, Aho teaches that some procedures are macro-expanded in the body of the source code. Therefore, the annotation functions of Kaneshiro can be thought of as macro-expanded, and hence, macros (Page 428, lines 1-5).

In regard to Claim 7, the 'name' argument, provided to a number of the annotation functions taught by Kaneshiro, is needed to produce annotation information according to the input argument (Column 8, lines 57-67).

In regard to Claim 9, Aho teaches macro expansion, which places and compilation time the body of a macro function that exists in the source code. Therefore, computer executable instructions associated with the annotation function is generated and placed into the source code at compile time (Page 428, lines 1-5). Aho describes a symbol table that the compiler creates and uses to keep track of variables and functions in the source code. The symbol table keeps track of a variety of arguments of a procedure (Page 11, lines 3-7), as well as the procedure name, so the computer executable instructions that represent the annotation function will, naturally, be associated with the annotation information as well.

In regard to Claim 10, Kaneshiro teaches generating annotation information (Column 5, lines 25-28), said information associated with a function. Kaneshiro teaches that the annotation information is gathered from annotation functions (Table 1), and the information gathered is associated with a function in that annotation functions are instrumented into the code at the start and end of a function in the code to gather data on the particular function. Kaneshiro does not teach inlining the annotation function in the second function. Aho, however, does teach in-line expansion of procedures (Page 428, lines 1-5). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize a method of annotating a computer program and gathering annotation information from the annotations, as taught by Kaneshiro, where the annotations are functions inlined in other functions of the source code, as taught by

Aho, since inlining decreases the overhead associated with calling a function and retrieving the function from another place in the source code.

In regard to Claim 11, the annotation information as taught by Kaneshiro, represents annotation functions in a symbol table as taught by Aho. The compiler uses the annotation information in the symbol table to run the annotation functions that collect debug information at execution time (Column 5, lines 25-28 and Column 5, lines 56-59).

In regard to Claim 13, Kaneshiro teaches that the annotation representations are functions inserted into the code (Column 5, lines 65-67) and the functions contain parameters (Table 1).

In regard to Claim 14, Kaneshiro teaches annotating computer source code using intrinsic function calls in the source code (Column 5, lines 65-67).

In regard to Claim 15, Kaneshiro teaches the method of Claim 14 and further teaches generating annotation information from the intrinsic function call (Column 5, lines 25-28 and Column 5, lines 56-59) and emitting annotation information into a computer file (Column 21, lines 39-41). Kaneshiro does not teach generating a symbol having string parameters of the function call and emitting the symbol to a symbol table. Aho, however, describes a symbol table that the compiler creates and uses to keep track of variables and functions in the source code. The symbol table keeps track of a variety of parameters of a procedure (Page 11, lines 3-7). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to annotate a computer source code with an intrinsic function call, generate annotation information from this function call, and store this information in an output file, as taught by Kaneshiro, where a symbol having string parameters is created for the function and stored on a

symbol table, as taught by Aho, since this allows the compiler to keep track of names generated in the computer program.

In regard to Claim 17, Claim 17 corresponds directly with Claim 9 and is rejected for the same reasons as Claim 9.

In regard to Claim 18, Claim 18 corresponds directly with Claim 4 and is rejected for the same reasons as Claim 4.

In regard to Claim 28, Claim 28 corresponds to Claim 14 and 15 and is rejected for the same reasons as Claim 14 and 15. Claims 38 and 41 correspond directly with Claim 28 and are rejected for the same reason as Claim 28.

In regard to Claims 29 and 30, Claims 29 and 30 correspond directly with Claim 15 and are rejected for the same reasons as Claim 15.

In regard to Claim 31, Kaneshiro teaches that profile functions are annotated into the source code (Column 5, lines 65-67). The profile functions can be thought of as software components.

In regard to Claim 32, Claim 32 corresponds with Claim 6 and is rejected for the same reasons as Claim 6.

In regard to Claim 42, Claim 42 corresponds with Claim 1 and is rejected for the same reasons as Claim 1.

5. Claim 12 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kaneshiro et al. (U.S. Patent Number 5,950,003) in view of "Compilers: Principles, Techniques, and Tools" by Alfred Aho (hereinafter Aho) and further in view of Shridhar (U.S. Patent Number 5,815,714).

In regard to Claim 12, the combination of Kaneshiro and Aho teach the method of Claim 11, but do not teach that the predetermined information comprises command line options. Shridhar, however, does teach generating debug information using command line options (Abstract). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to generate debug information from predetermined information as taught by Kaneshiro and Aho, where the predetermined information comprises command line options, as taught by Shridhar, since this allows the output of the debug information to be easily specified and controlled at execution time.

6. Claims 21, 27, 34, and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Buzbee (U.S. Patent Number 5,815,720) in view of Kaneshiro et al. (U.S. Patent Number 5,950,003).

In regard to Claim 21, Buzbee teaches the method of Claim 19, but does not teach that the annotation information is generated by a function call having at least one string parameter. Kaneshiro however, does teach attaining annotation information by means of a function call (Column 5, lines 25-28) with at least one string parameter (Table 1). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to control the execution of a computer analysis tool using annotation information as taught by Buzbee, where the annotation information is generated by a function call containing parameters, since function calls are a much easier and much more compact, and hence neater way of running multiple lines of code in a source program. Claims 27, 34, and 36 correspond to Claim 21 and are rejected for the same reason as Claim 21.

7. Claims 39 and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Compilers: Principles, Techniques, and Tools" by Alfred Aho (hereinafter Aho) in view of Kaneshiro et al. (U.S. Patent Number 5,950,003).

In regard to Claim 39, Aho teaches a data structure that contains information corresponding to a function in a computer program. Aho describes a symbol table that the compiler creates and uses to keep track of variables and functions in the source code. The symbol table keeps track of a variety of arguments of a procedure (Page 11, lines 3-7), as well as the procedure name. Aho does not teach that the functions are annotation functions, and that the information is annotation information. Kaneshiro, however, does teach using annotation functions (Table 1) to collect annotation information (Column 5, lines 25-28). Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to store a data structure comprising information corresponding to a function in a source code as taught by Aho, where the function is an annotation function, as taught by Kaneshiro, since this allows the compiler to keep track of names generated in the computer program.

In regard to Claim 40, Aho teaches that the symbol table keeps track of a variety of arguments of a procedure (Page 11, lines 3-7).

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

Bennett et al. (U.S. Patent Number 6,049,666).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kenneth A Gross whose telephone number is (703) 305-0542. The examiner can normally be reached on Mon-Fri 7:30-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Gregory A Morse can be reached on (703) 308-4789. The fax phone numbers for the organization where this application or proceeding is assigned are (703) 746-7239 for regular communications and (703) 746-7240 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

KAG
April 1, 2003

GREGORY MORSE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100